In the Claims:

1. (Currently amended) ~~In a~~ A system for providing access control management to electronic data, wherein the electronic data is structured in a format that provides restricted access to the electronic data therein, ~~the format~~ comprising:

a client module configured to generate a header ~~including~~ comprising encrypted security information as to who and how a file including the electronic data can be accessed[[;]], and configured to generate an encrypted data portion ~~including~~ comprising the file encrypted with a file key according to a predetermined cipher scheme[[;]], ~~and~~

wherein the header is ~~attached~~ coupled to the encrypted data portion to generate a secured file.

2. (Currently amended) The ~~format~~ system as recited in Claim 1, wherein the security information in the header of the secured file facilitates the restricted access to the file.

3. (Currently amended) The ~~format~~ system as recited in Claim 1, wherein the security information is encrypted with a user key associated with a user.

4. (Currently amended) The ~~format~~ system as recited in Claim 3, wherein the user is a member selected from a group consisting of a human user, a software agent, a device and a group of users; and wherein the user is granted access privilege to access the file.

5. (Currently amended) The ~~format~~ system as recited in Claim 4, wherein the security information ~~includes~~ comprises the file key and access rules to the restricted access to the file.

6. (Currently amended) The ~~format~~ system as recited in Claim 5, wherein the file key is retrieved to decrypt the encrypted data portion in the secured file when the access privilege of the user is within access permissions by the access rules.

7. (Currently amended) The ~~format~~ system as recited in Claim 6, wherein the access rules are expressed in a markup language.

8. (Currently amended) The ~~format~~ system as recited in Claim 7, wherein the markup language is Extensible Access Control Markup Language.

9. (Currently amended) The ~~format~~ system as recited in Claim 7, wherein the markup language is selected from a group consisting of HTML, XML and SGML.

10. (Currently amended) The ~~format~~ system as recited in Claim 1, wherein the secured file is configured to have a file extension identical to what the file originally has so that an application designated to access the file can be executed to access the secured file.

11. (Currently amended) The ~~format~~ system as recited in Claim 10, wherein the security information ~~includes~~ comprises a flag to the application that the secured file being accessed can not be accessed as it normally does.

12. (Currently amended) The ~~format~~ system as recited in Claim ~~10~~ 11, wherein the flag is configured to be placed in a position of the secured file so that the flag will be accessed first when the secured file is accessed by the application.

13. (Currently amended) The ~~format~~ system as recited in Claim 10, wherein the security information ~~includes~~ comprises the file key and access rules, the access rules

controlling who and how the secured file can be accessed, and wherein the security information in the header is organized in such a way that the application is paused, upon detecting that the secured file is being accessed, for an access control module to determine whether a user requesting the secured file has proper access privilege to do so with respect to the access rules in the security information.

14. (Currently amended) The ~~format~~ system as recited in Claim 13, wherein the access control module ~~operating~~ operates in a path through which the secured file is confined to be loaded into the application.

15. (Currently amended) The ~~format~~ system as recited in Claim 1, wherein the file key is a symmetric cipher key.

16. (Currently amended) The ~~format~~ system as recited in Claim 1, wherein the file is one or more of a document, a multimedia file, a set of dynamic or static data, a sequence of executable code, an image and a text.

17. (Currently amended) ~~In a~~ A system for providing access control management to electronic data, wherein the electronic data is structured in a format that provides restricted access to the electronic data therein, ~~the format~~ comprising:

a client module configured to generate a header including an encrypted file key ~~encrypted~~ and a rule block having N encrypted segments, each of the N encrypted segments including a set of access rules facilitating the restricted access to a file including the electronic data, wherein N >=1[[;]] and an encrypted data portion including the electronic data encrypted according to a predetermined cipher;

wherein the header is ~~attached~~ coupled to the encrypted data portion to generate a secured file[[;]], and the file key can be retrieved to decrypt the encrypted data portion only when the access rules in one of the N encrypted segments are measured successfully against access privilege associated with a user accessing the secured file.

18. (Currently amended) The ~~format~~ system as recited in Claim 17, wherein the header further ~~includes~~ comprises a user block having user information identifying who can access the secured file.

19. (Currently amended) The ~~format~~ system as recited in Claim ~~18~~ 17, wherein each of the N encrypted segments of the rule block ~~includes~~ comprises policies on how the secured file can be accessed.

20. (Currently amended) The ~~format~~ system as recited in Claim 18, wherein the user block includes N encrypted segments, each including the file key.

21. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein each of the N encrypted segments of the user block corresponds to one of the N encrypted segments of the rule block.

22. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein each of the N encrypted segments of the user block further ~~includes~~ comprises a user identification identifying who can access the secured document.

23. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein each of the N encrypted segments of the user block further ~~includes~~ comprises cipher information about the predetermined cipher to facilitate a decryption process of the encrypted data portion with the file key.

24. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein the access rules in each of the N encrypted segments of the rule block determine at least an action with which the secured document can be accessed by a user associated with one of the N encrypted segments of the user block.

25. (Currently amended) The ~~format~~ system as recited in Claim 24, wherein the action ~~includes~~ comprises one or more of commands: open, export, read, edit, play, listen to, print or forward and attach.

26. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein the access rules in each of the N encrypted segments of the rule block are expressed in a ~~marked-up~~ markup language.

27. (Currently amended) The ~~format~~ system as recited in Claim 26, wherein the markup language is Extensible Access Control Markup Language.

28. (Currently amended) The ~~format~~ system as recited in Claim 26, wherein the markup language is selected from a group consisting of HTML, XML and SGML.

29. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein the N encrypted segments of the user block are respectively encrypted with the file key.

30. (Currently amended) The ~~format~~ system as recited in Claim 29, wherein an authorized user associated with one of the encrypted segments of the user block can view the access rules of each of the N encrypted segments of the rule block when access privilege of the authorized user is measured successfully with the access rules in one of the N encrypted segments in the rule block associated with the authorized user.

31. (Currently amended) The ~~format~~ system as recited in Claim 30, wherein the authorized user can update the access rules of each of the N encrypted segments of the rule block.

32. (Currently amended) The ~~format~~ system as recited in Claim 20, wherein the N encrypted segments of the user block remain encrypted every time the secured file is stored in a storage space.

33. (Currently amended) In a system for providing access control management to electronic data, wherein the electronic data is structured in a format that provides restricted access to the electronic data therein, a method for generating the format, comprising:

obtaining a file key;

encrypting the electronic data with the file key according to a predetermined cipher to produce an encrypted data portion; and

integrating a header comprising encrypted security information with the encrypted data portion to generate a secured file, wherein the encrypted security information ~~includes~~ comprises the file key and access rules to control the restricted access to the electronic data in the secured file.

34. (Currently amended) The method of Claim 33, wherein the encrypted security information ~~includes~~ comprises user information as to who can access the secured file.

35. (Currently amended) The method of Claim 34, wherein the encrypted security information ~~is encrypted and~~ can only be decrypted by a user key associated with a user identified in the user information in the encrypted security information.

36. (Original) The method of Claim 34, wherein the user is a member selected from a group consisting of a human user, a software agent, a device and a group of users; and wherein the user is granted access privilege to access the secured file.

37. (Original) The method of Claim 36 further comprising obtaining the access rules from either a default setting for a file place in which the secured file is to be placed or a manual setting in accordance with access privilege associated with a user who is creating the secured file.

38. (Original) The method of Claim 33, wherein the obtaining of the file key comprises:

if the secured file is newly generated,

generating the file key from the predetermined cipher; and

if the secured file is being stored in a storage place,

retrieving the file key from a memory store; and

deleting the file key from a memory store as soon as the secured file is stored in the storage place.